Assignment Zero: C++ Warmup

CS314 Operating Systems

Write a program in C++ that implements a smart memory manager for a graph that has no memory leaks.

Assemble a 10x10 grid of nodes connected by edges. Use a C++ class or struct to represent a node. Use C++ pointers to represent an edge. Edges are bidirectional so you will need two pointers (one from node A to node B and one from node B to node A) to represent an edge.

A node will have two (corner), three (grid edge, non-corner), or four edges (non-corner, non-grid edge) depending on its location in the grid.

Nodes will also have an identifier.

Delete a random node by generating a random identifier, and traversing the grid until you locate the node with that has that identifier.

When deleted node sits between other nodes, you will need to make sure that the other nodes are reconnected around the deleted node

```
A B C

D E F

G H I
```

For example if node E is deleted, then you need to connect D-F and B-H

The program terminates when there are no more nodes to delete.

NOTE: You must use the new operator to construct and assemble nodes in the grid and you must use the delete operator to eliminate nodes in the grid.

NOTE: You can read about memory leak detection schemes here -> https://gcc.gnu.org/onlinedocs/libstdc++/manual/debug.html

NOTE: I'll be using valgrind to check for memory leaks with this command:

```
valgrind --leak-check=yes ./assignment0
```

DELIVERABLES (Please follow these directions precisely… it saves me a lot of time when I grade your assignment):

Upload to elearn the following:

- An ELECTRONIC document describing how to run the program you created. Call this document README.TXT.
- These files should be placed in a directory called "`<username>assignment0`".
- Use the tar command to place all the files in a single file called "`<username>assignment0.tar`". Assuming you are in the directory "`<username>assignment0`" do the following:
    - Goto the parent directory: `cd ..`
    - tar the files: `tar -cvf <username>project0.tar ./<username>assignment0`
    - Verify the files have been placed in a tar file: `tar -tvf <username>assignment0.tar`
    - Compress the files using gzip: `gzip <username> assignment0.tar`
    - Verify that the gzipped file exists: `ls <username> assignment0.tar.gz`
- Here's a screen snapshot (just replace assignment0 with project1) of these commands: